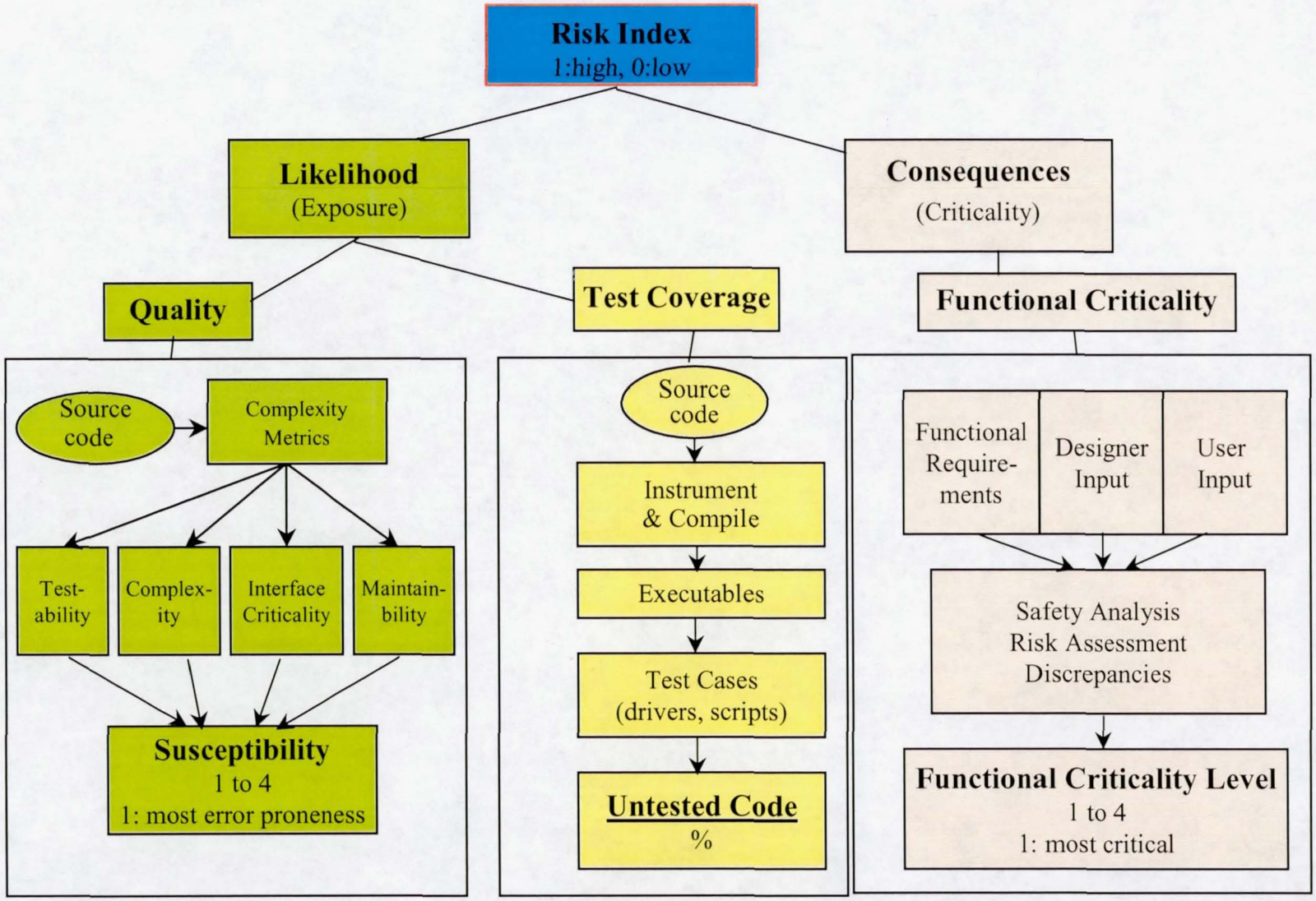
A Quantitative Software Risk Assessment Model

Alice Lee

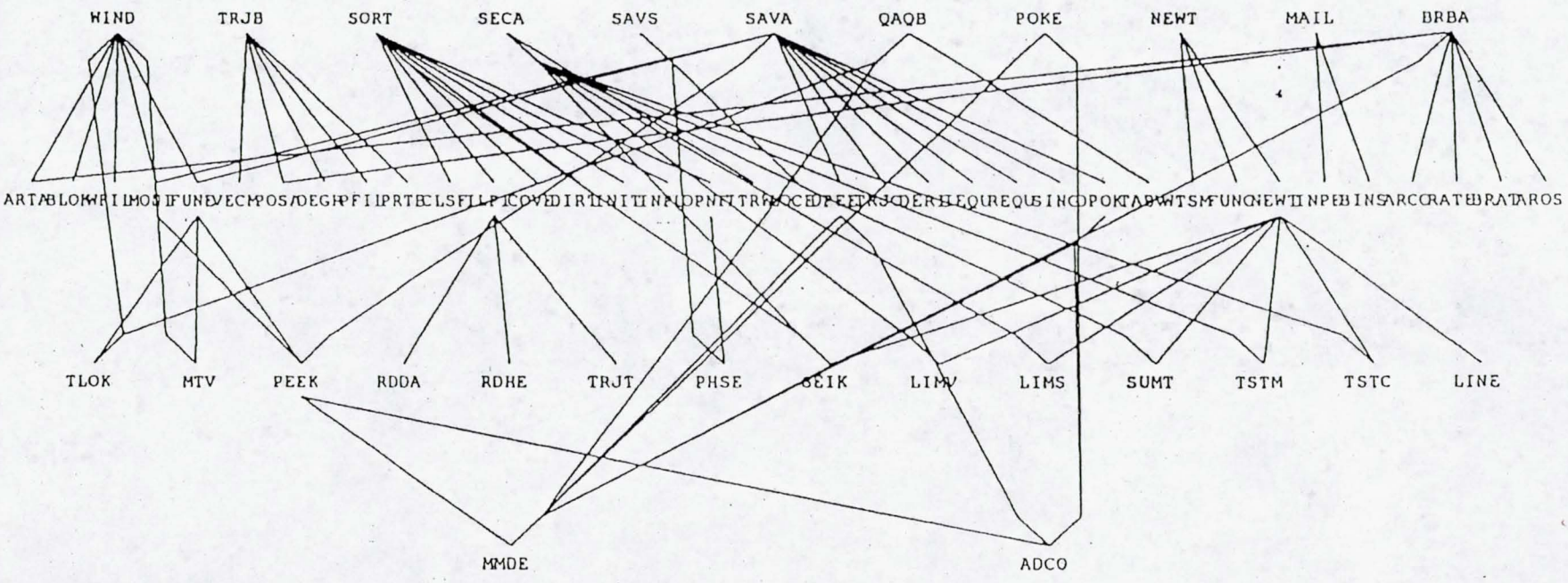
A Risk Model


Risk Index
1:high, 0:low
Likelihood
(Exposure)
Consequences
(Criticality)
Quality
Test Coverage
Functional Criticality
Source code
Complexity Metrics
Test-ability
Complex-ity
Interface Criticality
Maintain-bility
Susceptibility
1 to 4
1: most error proneness
Source code
Instrument & Compile
Executables
Test Cases
(drivers, scripts)
Untested Code
%
Functional Require-ments
Designer Input
User Input
Safety Analysis
Risk Assessment
Discrepancies
Functional Criticality Level
1 to 4
1: most critical

Risk Index: A Sample Result

Call Grph Number	PEG_Task Components	Overall Susceptibility (1 = bad)	Total Test Coverage %	Functional Criticality (1=most crit)	Risk Index (0 to 1, 1=high)	S/W Reliability
7	Current_Acc	4	72%	1	0.07	
8	CALC_VRATIO	4	73%	1	0.0675	
9	Time_To_Go	4	89%	1	0.0275	
10	Thrust_Integral	3.5	100%	1	0	
11	Reference_Thrust_Vectors	4	100%	1	0	
12	Util_Midval	4	70%	1	0.075	
13	UTIL_V_Magnitude	3.5	0%	1	0.375	
14	Gmd_Pred_Eq_Zero	4	90%	1	0.025	
15	UTIL_MT	4	0%	1	0.25	
16	UTIL_M1_Times_M2	3.5	0%	1	0.375	
17	UTIL_M_Times_V	3.5	0%	1	0.375	
18	UTIL_MT_Times_V	3	0%	1	0.5	
19	UnitVector	4	93%	1	0.0175	
20	H_Ellipsoid	4	96%	1	0.01	
21	Alt_Ge_400k	4	0%	1	0.25	
22	Alt_Lt_400K	4	0%	1	0.25	
23	Alt_lt_600k	3	0%	1	0.5	
24	Alt_Ge_600k	4	0%	1	0.25	
25	Accel_Drag	3	0%	1	0.5	
26	Drag_Accel	3	0%	1	0.5	
27	Drag_Accel_Bypass	4	0%	1	0.25	
28	delta_accel_imu	1	0%	1	1	
29	no_delta_accel	4	0%	1	0.25	
30	Gmd_Pred_Not_Zero	3	0%	1	0.5	
31	Nsteps_Stepsz	3	86%	1	0.07	
32	Nstep_Do_Loop	1	23%	1	0.77	
33	Entry_Precise_Predictor	3	19%	1	0.405	
34	Burnout_State_Vector_Pred	3	42%	1	0.29	
35	V_Magnitude	3.5	100%	1	0	
36	Send_Unit_Vector	4	100%	1	0	
37	Send_Zero_Vector	4	100%	1	0	
38	UTIL_V_Unitize	3	68%	1	0.16	

Example Result of Complexity Study (1)



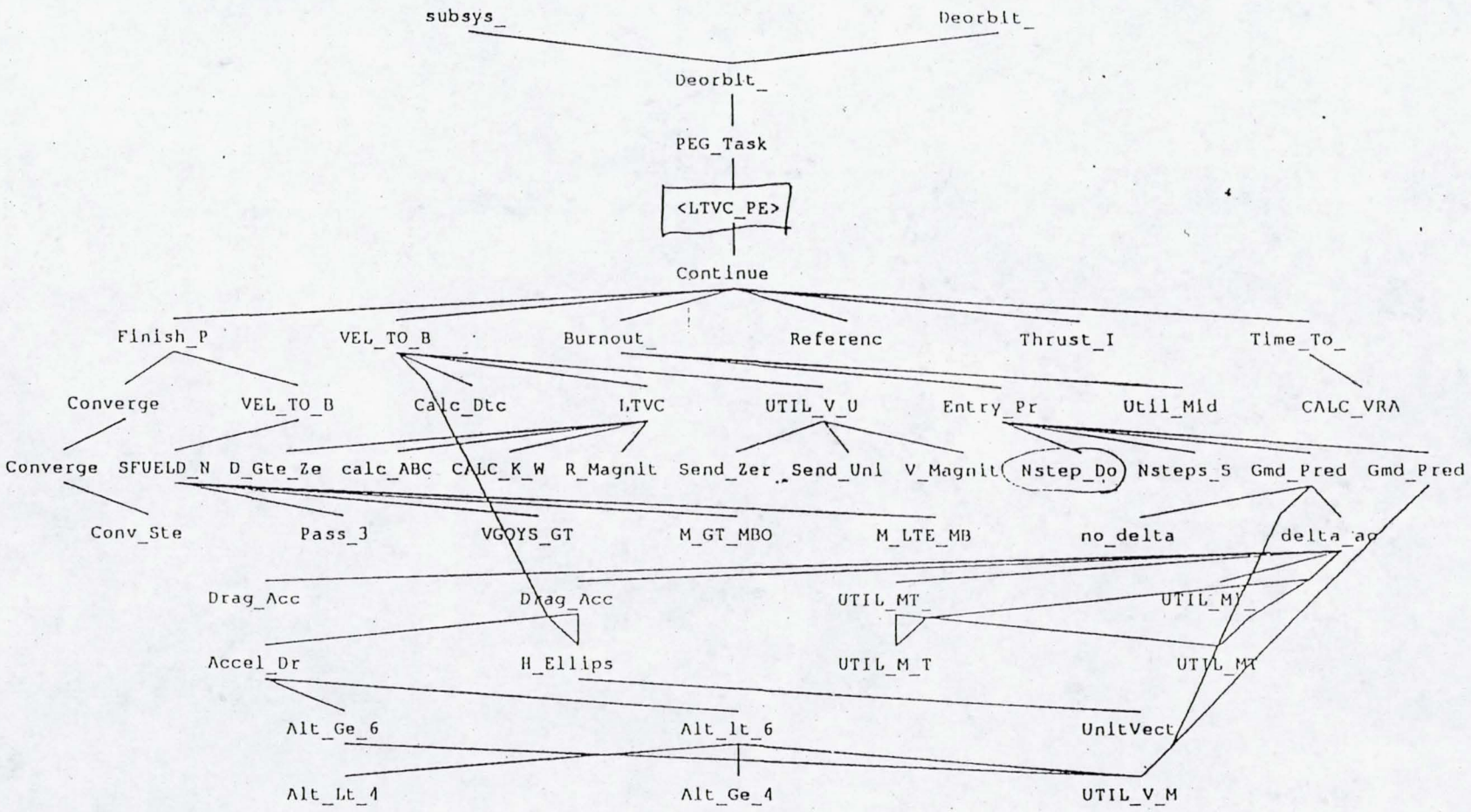
Application: newcode
Version: VERSION1

Language: FORTRAN

Call graph of root:

//

Example Result of Complexity Study (2)



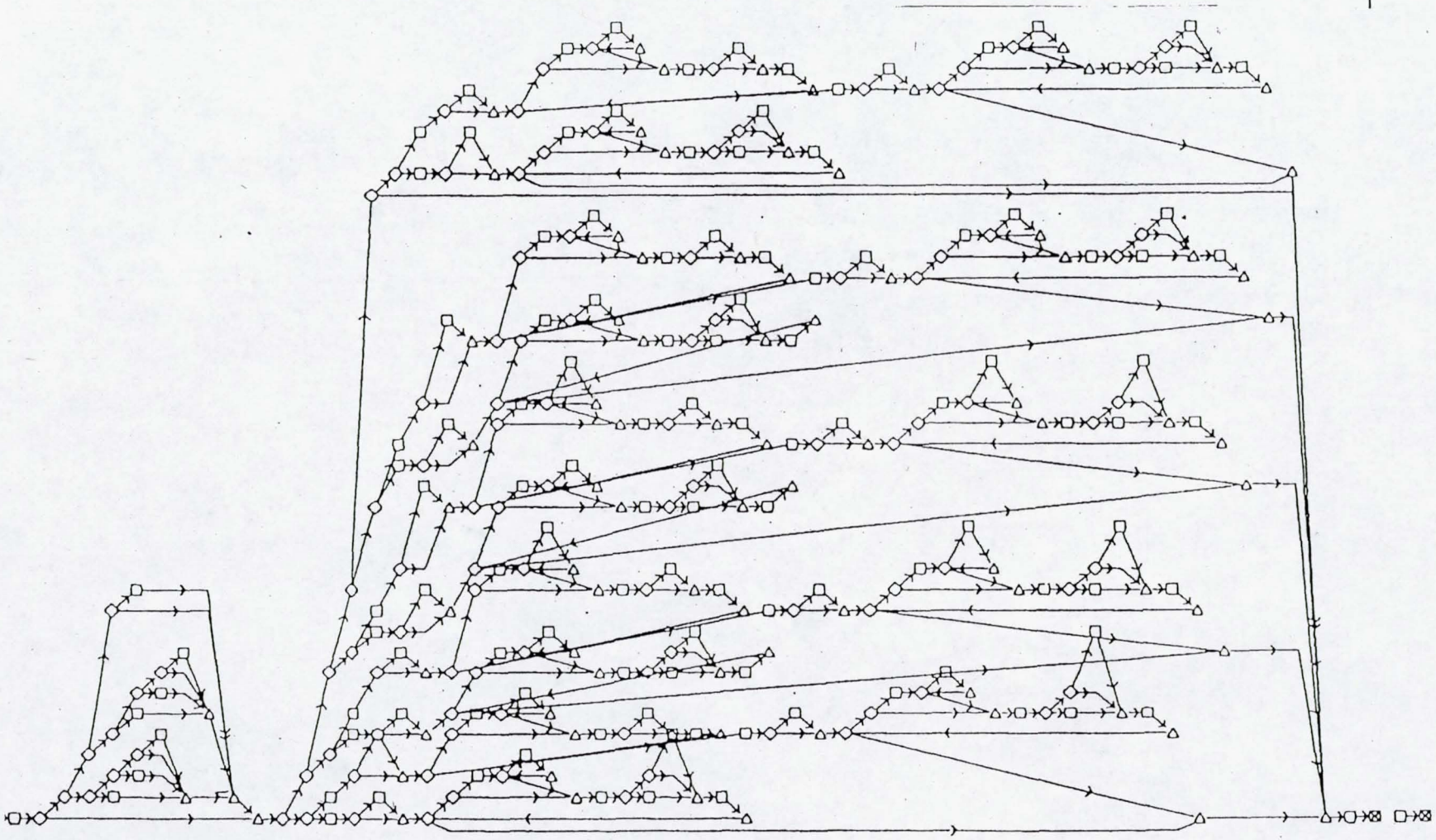
Application: SOURCE
Version: CURRENT_VERSION

Language: C

Calling/called components graph of the root :

LTVC_PEG

Example Result of Complexity Study (3)



g) = 106

plication: sorter
rsion: VERSION1

Language: FORTRAN

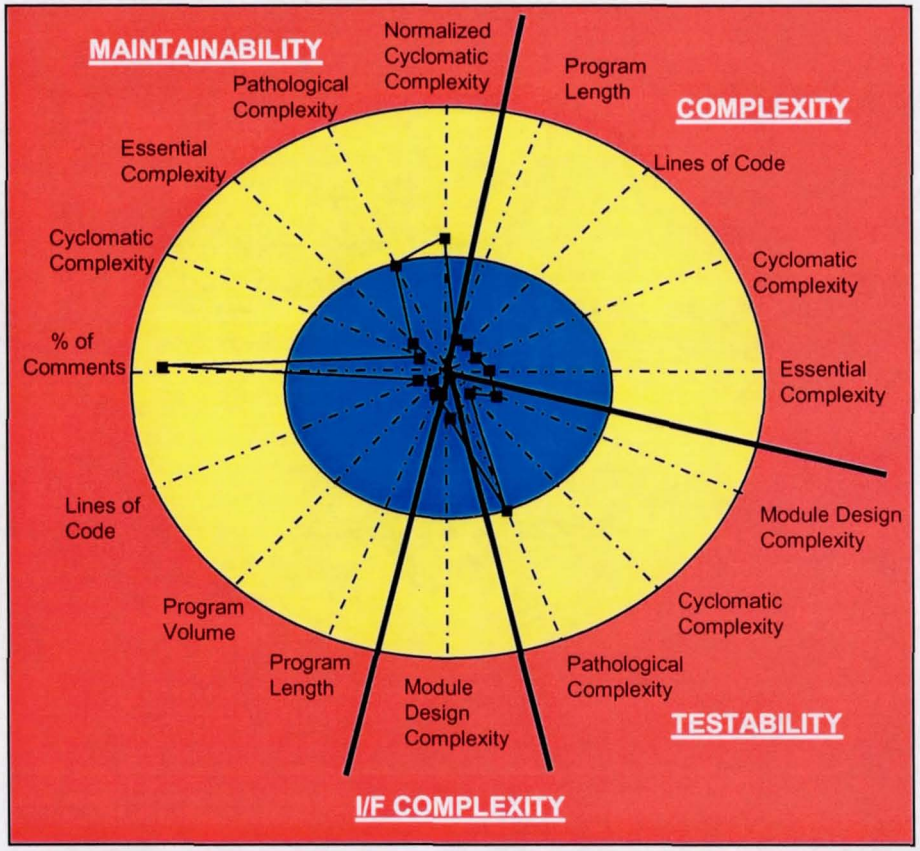
Control graph of component:

EVTPRT

Example - A "Green" module (Least Error Proneness)

Susceptibility Level = 1.1

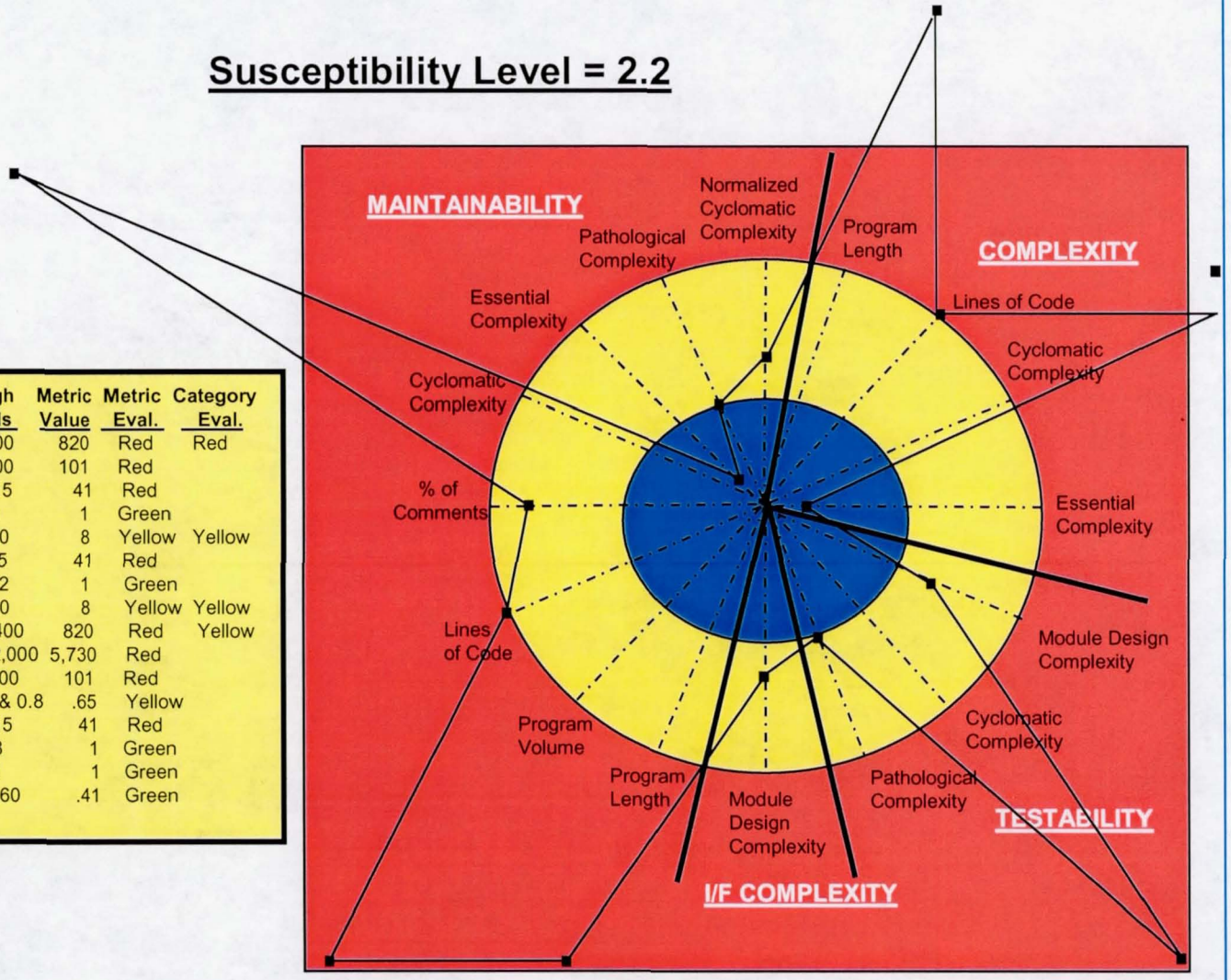
Category	Metric	Low & High Thresholds	Metric Value	Metric Eval.	Category Eval.
Complexity	Prog. Length	300 & 400	21	Green	Green
	Lines of Code	60 & 100	6	Green	
	Cyclom. Complex.	10 & 15	2	Green	
	Essent. Complex.	4 & 8	1	Green	
Testability	Design Complex.	7 & 10	2	Green	Green
	Cyclom. Complex.	10 & 15	2	Green	
	Pathol. Complex.	1 & 2	1	Green	
I/F Complex.	Design Complex.	7 & 10	2	Green	Green
Maintainab.	Prog. Length	300 & 400	21	Green	Green
	Prog. Volume	*1,500 & 2,000*	82	Green	
	Lines of Code	60 & 100	6	Green	
	1 - % of Comments	0.2 & 0.8	.74	Yellow	
	Cyclom. Complex.	10 & 15	2	Green	
	Essent. Complex.	4 & 8	1	Green	
	Pathol. Complex.	1 & 2	1	Green	
	Normal. Cyclom. Complexity	.28 & .60	.33	Yellow	



Example - A "Yellow" module (Medium Error Proneness)

Susceptibility Level = 2.2

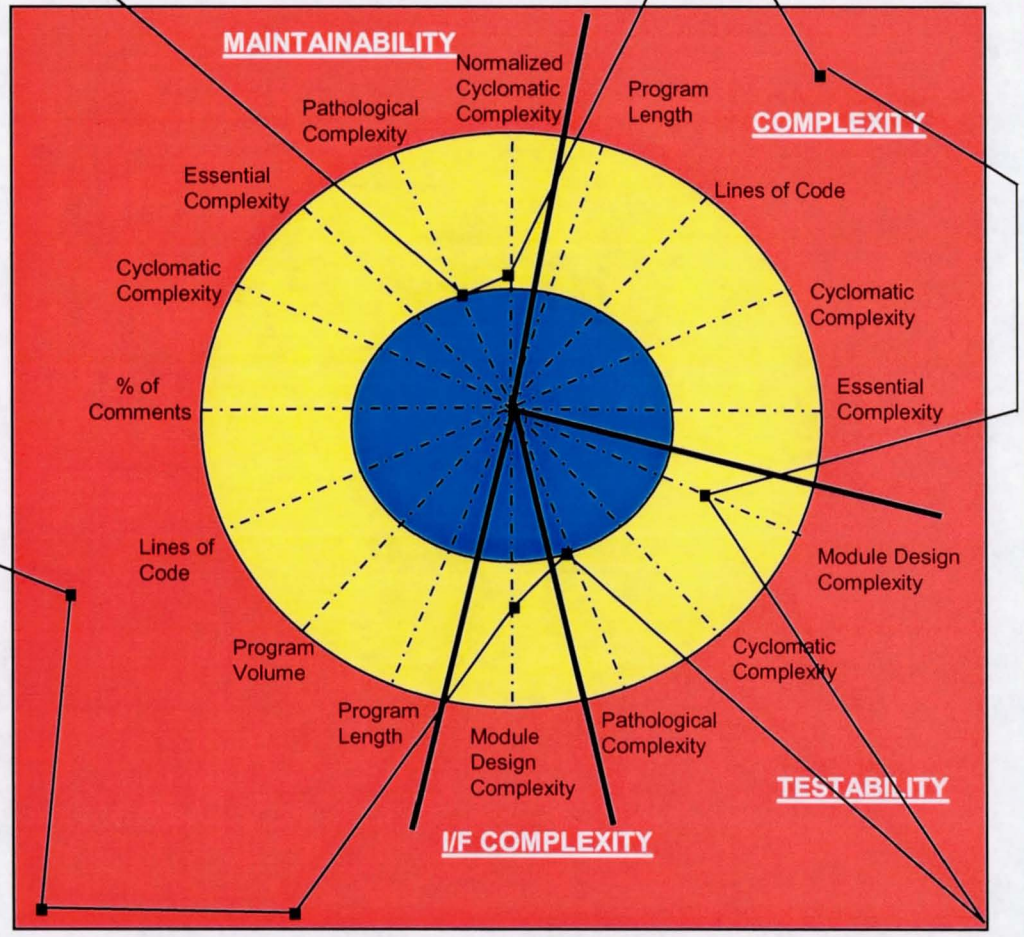
Category	Metric	Low & High Thresholds	Metric Value	Metric Eval.	Category Eval.
Complexity	Prog. Length	300 & 400	820	Red	Red
	Lines of Code	60 & 100	101	Red	
	Cyclom. Complex.	10 & 15	41	Red	
	Essent. Complex.	4 & 8	1	Green	
Testability	Design Complex.	7 & 10	8	Yellow	Yellow
	Cyclom. Complex.	10 & 15	41	Red	
	Pathol. Complex.	1 & 2	1	Green	
I/F Complex.	Design Complex.	7 & 10	8	Yellow	Yellow
Maintainab.	Prog. Length	300 & 400	820	Red	Yellow
	Prog. Volume	1,500 & 2,000	5,730	Red	
	Lines of Code	60 & 100	101	Red	
	1 - % of Comments	0.2 & 0.8	.65	Yellow	
	Cyclom. Complex.	10 & 15	41	Red	
	Essent. Complex.	4 & 8	1	Green	
	Pathol. Complex.	1 & 2	1	Green	
	Normal. Cyclom. Complexity	.28 & .60	.41	Green	



Example - A "Red" module (Most Error Proneness)

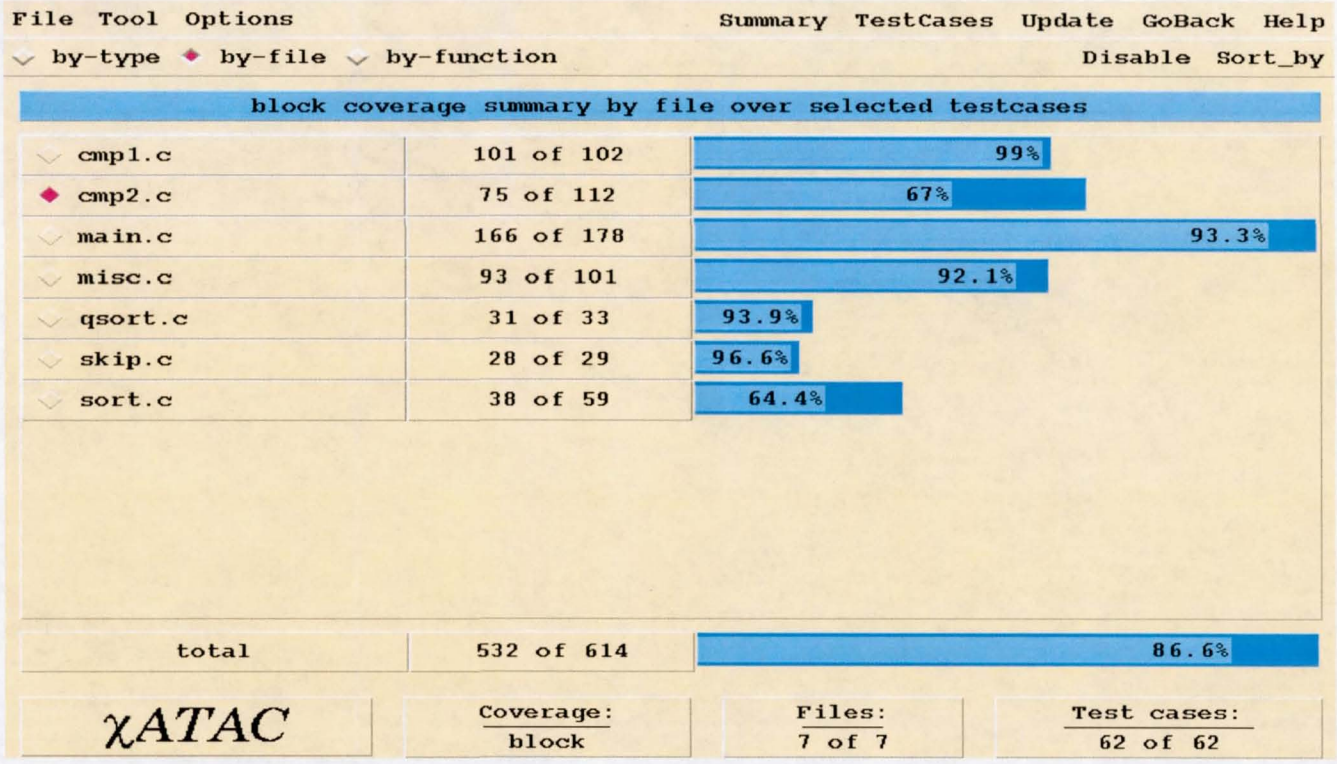
Susceptibility Level = 2.5

Category	Metric	Low & High Thresholds	Metric Value	Metric Eval.	Category Eval.
Complexity	Prog. Length	300 & 400	633	Red	Red
	Lines of Code	60 & 100	137	Red	
	Cyclom. Complex.	10 & 15	41	Red	
	Essent. Complex.	4 & 8	30	Red	
Testability	Design Complex.	7 & 10	8	Yellow	Yellow
	Cyclom. Complex.	10 & 15	41	Red	
	Pathol. Complex.	1 & 2	1	Green	
I/F Complex.	Design Complex.	7 & 10	8	Yellow	Yellow
Maintainab.	Prog. Length	300 & 400	633	Red	Yellow
	Prog. Volume	1,500 & 2,000	4,431	Red	/
Red					
	Lines of Code	60 & 100	137	Red	
	1 - % of Comments	0.2 & 0.8	.40	Yellow	
	Cyclom. Complex.	10 & 15	41	Red	
	Essent. Complex.	4 & 8	30	Red	
	Pathol. Complex.	1 & 2	1	Green	
	Normal. Cyclom. Complexity	.28 & .60	.30	Yellow	

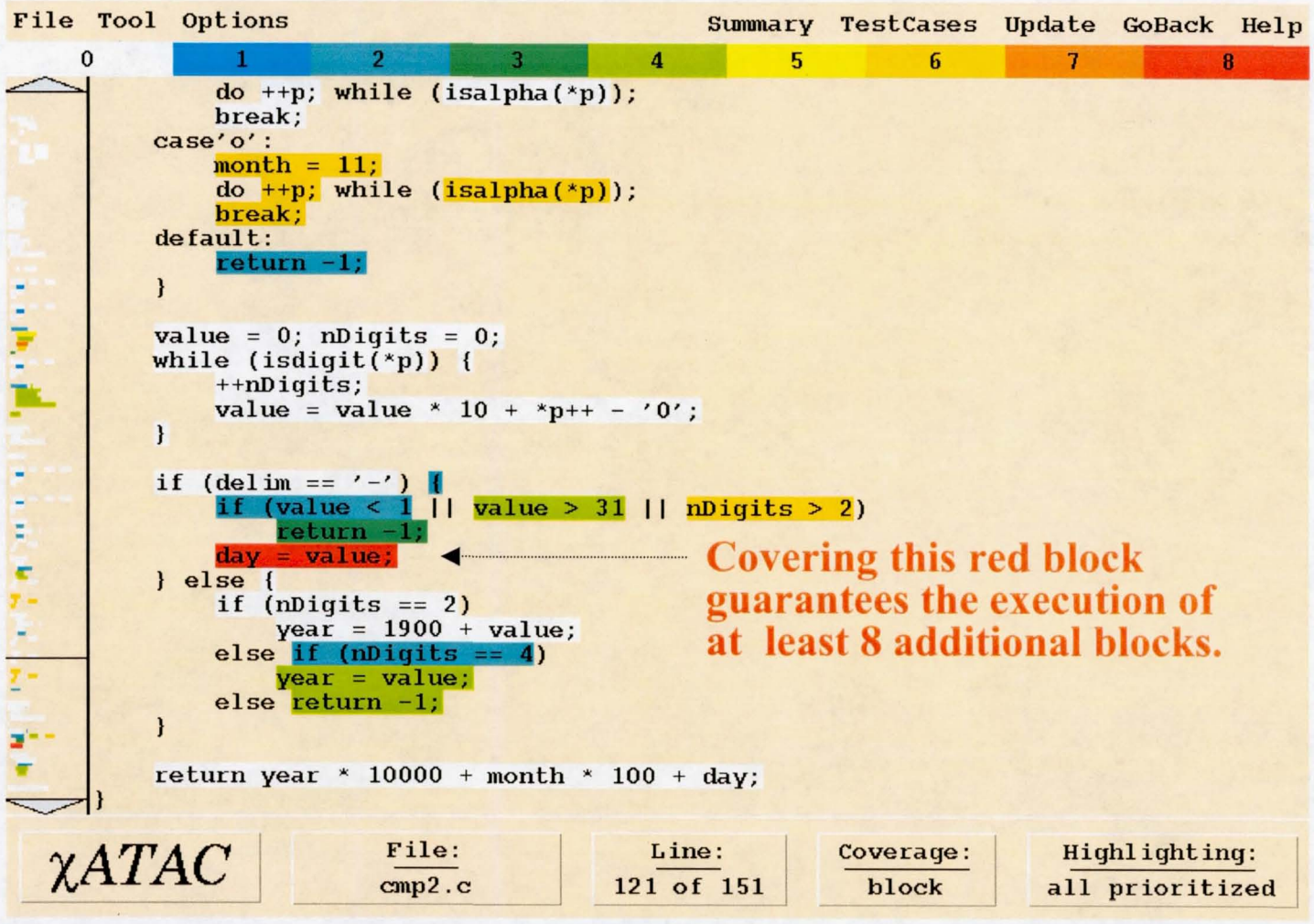


Example - Analyze Test Coverage of Code and Test Efficiency (1)

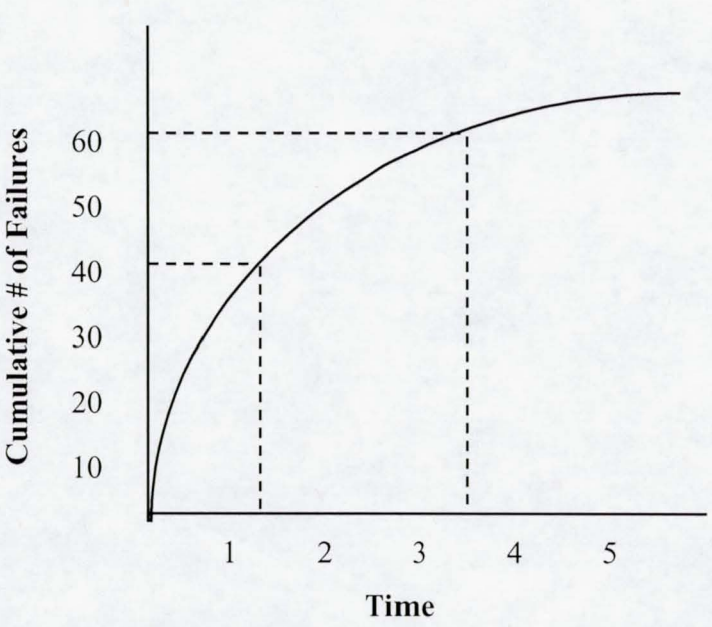
- *cmp2.c*, the most relevant function with date sorting, only has a 67% block coverage

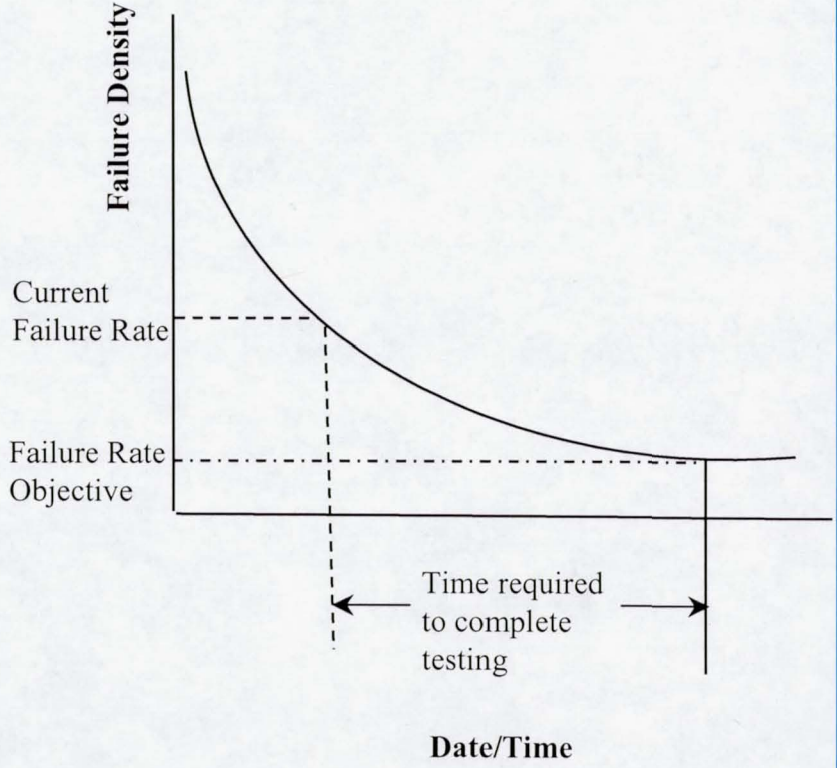


Example - Analyze Test Coverage of Code and Test Efficiency (2)



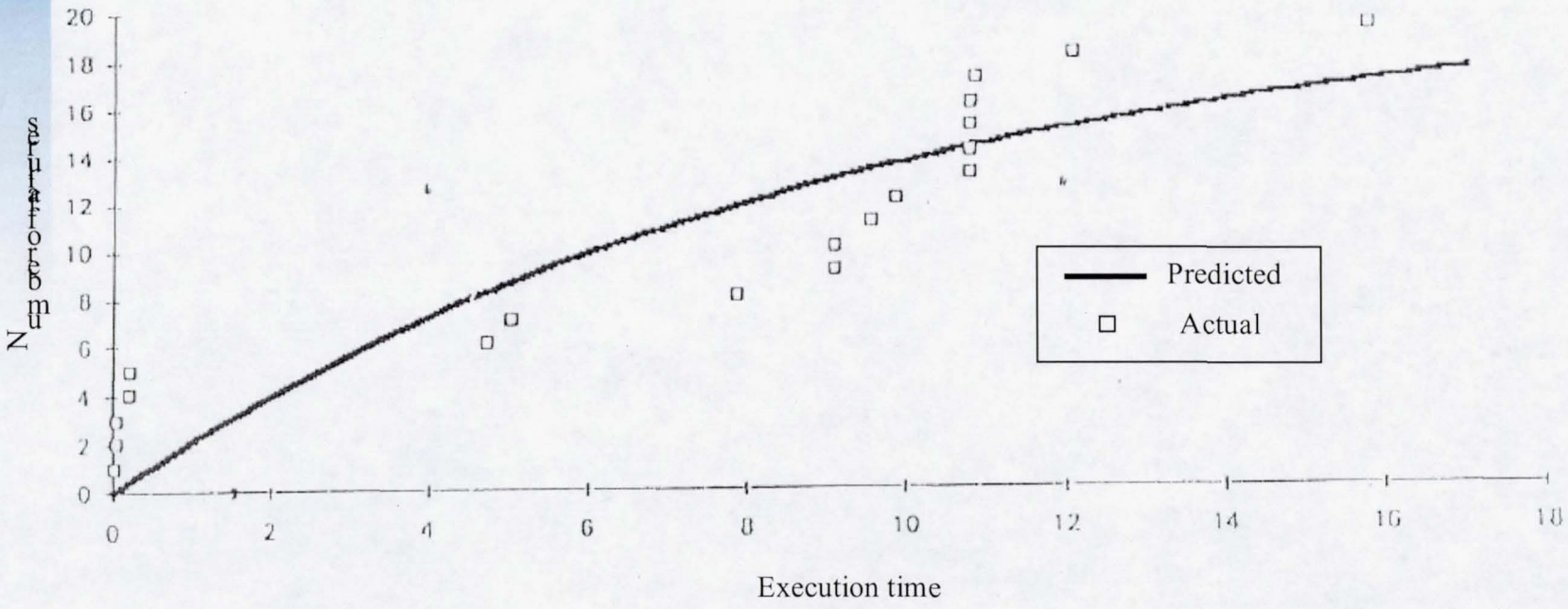
Software Reliability: Basic Concepts



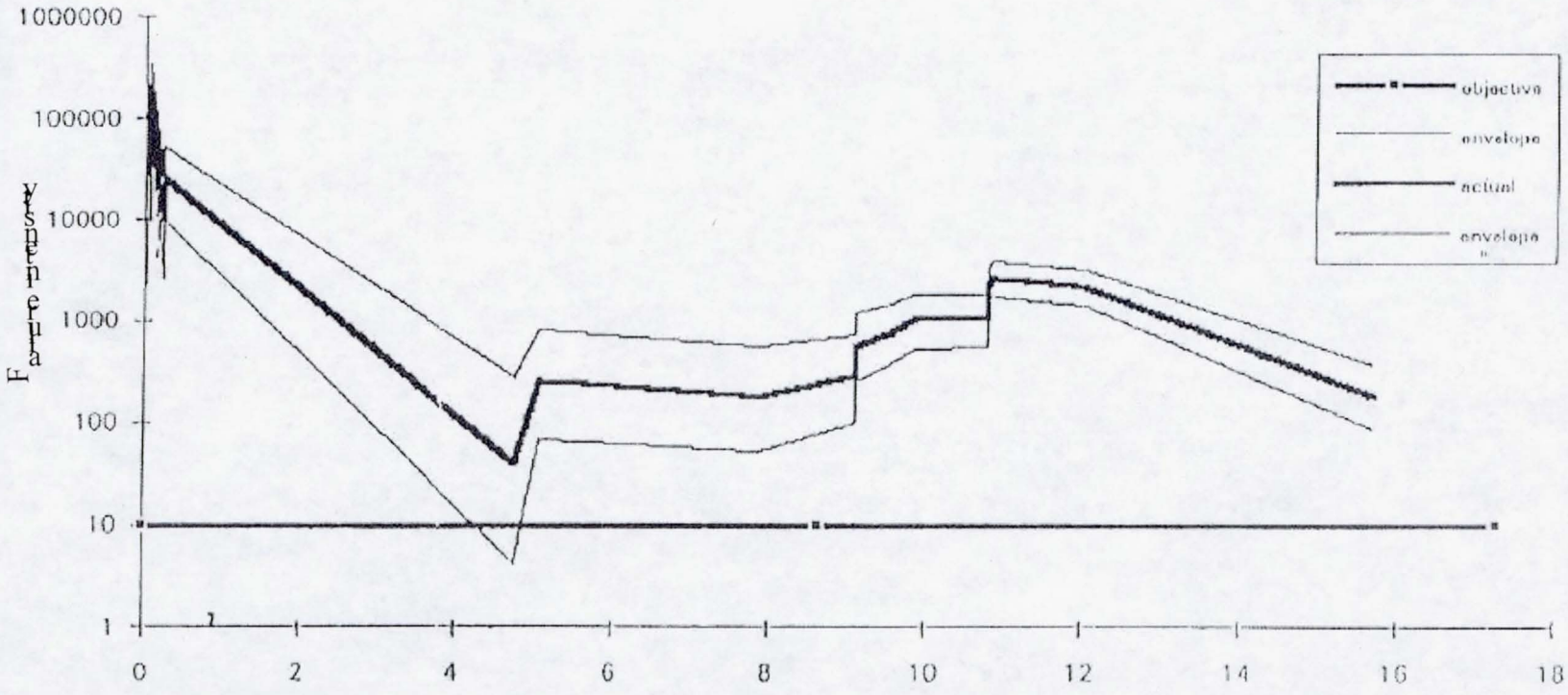


Software Reliability: Example (1)

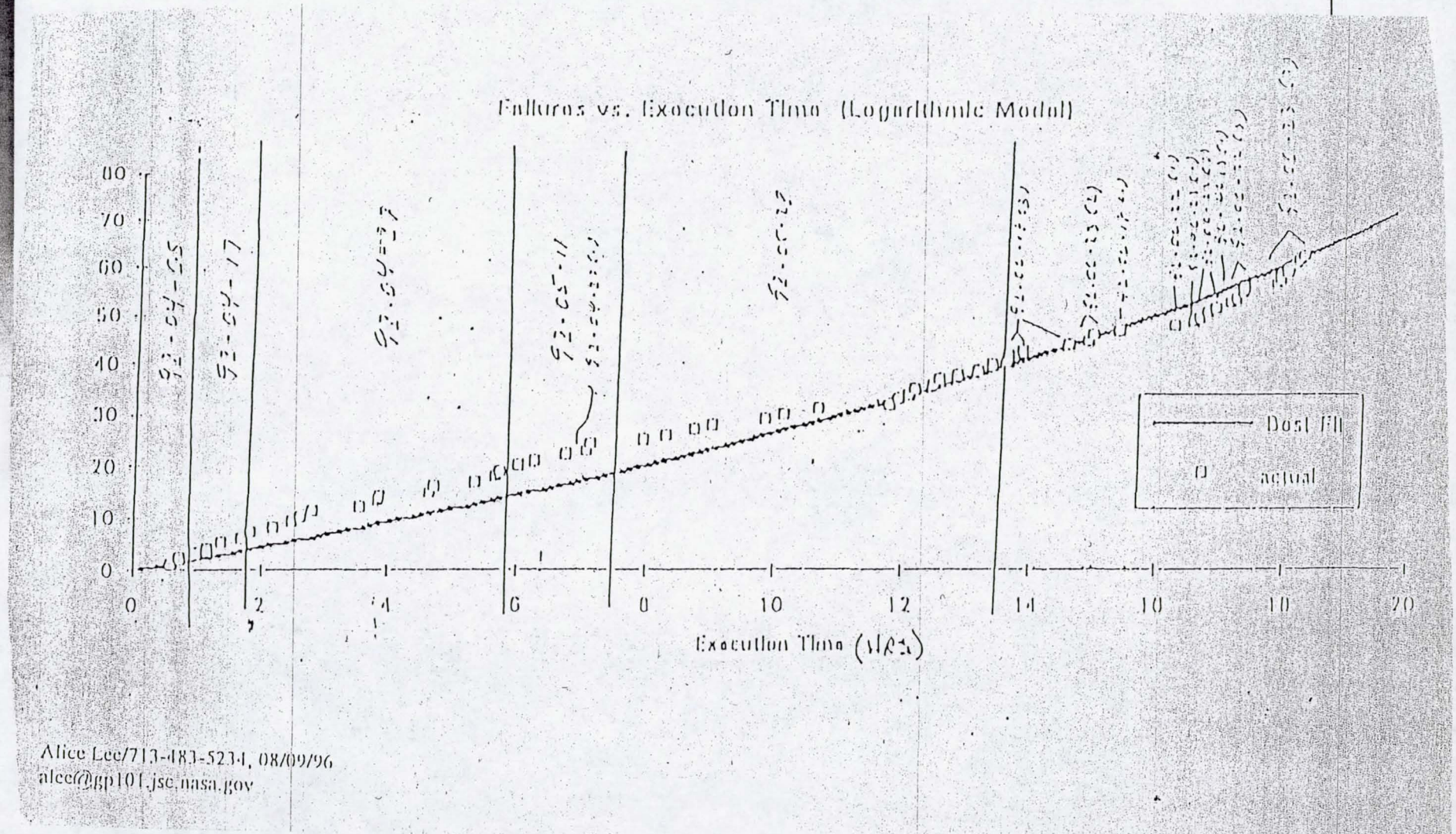
Musa's Basic Execution Time Model



Software Reliability: Example (2)

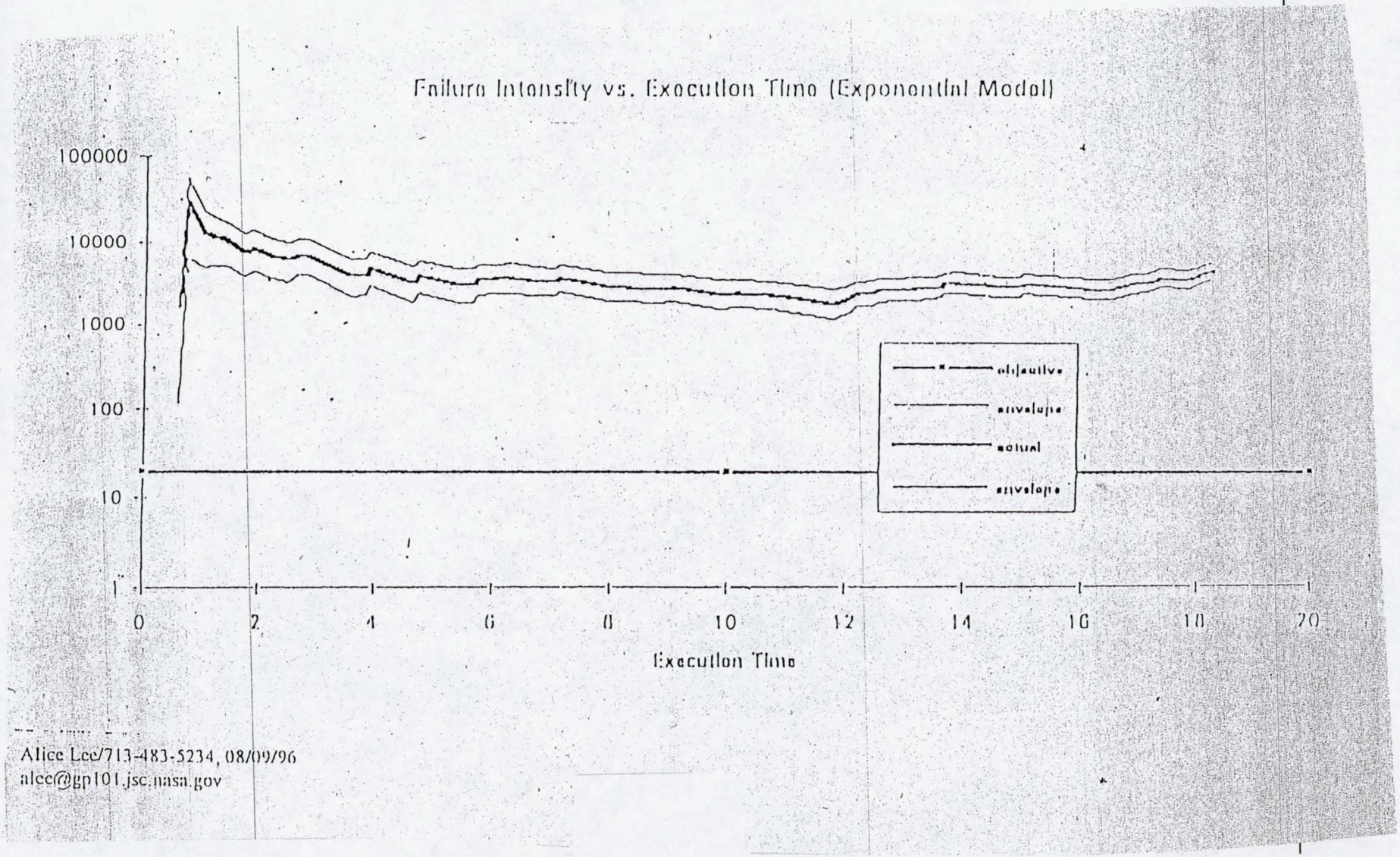


Software Reliability: Example (3)



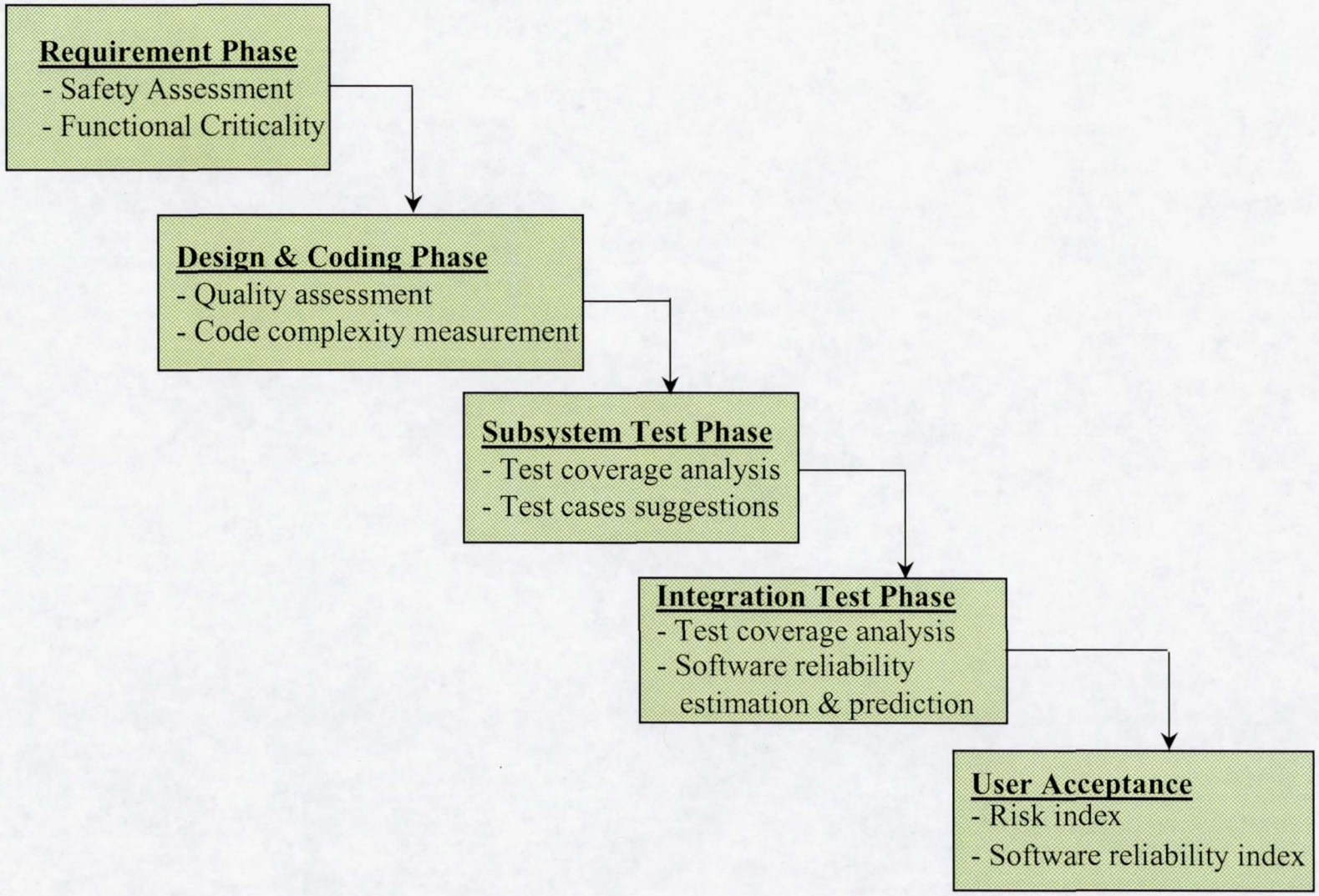
Alice Lee/713-483-5234, 08/09/96
alee@gp101.jsc.nasa.gov

Software Reliability: Example (4)



Alice Lee/713-483-5234, 08/09/96
alee@gp101.jsc.nasa.gov

Life Cycle Application of the Risk Model



Software Capability Maturity Level

Level	Characteristics
I. Initial	Ad hoc, few processes are defined
II. Repeatable	Basic project management processes are established to track cost, schedule, and functionality
III. Defined	Process for both management & engineering activities is institutionalized
IV. Managed	Process and product quality are quantitatively understood and controlled
V. Optimizing	Improvement is enabled by quantitative feedback from the piloting of innovative ideas and technologies